2

HSD-TR-89-010

NOISE AND SONIC BOOM IMPACT TECHNOLOGY

Initial Development of an Assessment System for Aircraft Noise (ASAN): System Design Strategy

Volume II of IV Volumes

Sanford Fidell
Nicolaas Reddingius
Michael Harris
B. Andrew Kugler

BBN Systems & Technologies Corporation
21120 Vanowen Street
Canoga Park, CA 91303

June 1989

Final Report for Period February 1987 - October 1988

Approved for public release; distribution is unlimited.

Noise and Sonic Boom Impact Technology Program
Human Systems Division
Air Force Systems Command
Brooks Air Force Base, TX 78235-5000

NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility nor any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise as in any manner construed, as licensing the holder, or any other person or corporation; or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

The Office of Public Affairs has reviewed this report and it is releasable to the National Technical Information Service (NTIS), where it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for publication.

ROBERT C. KULL, JR, Capt, USAF
NSBIT Program Manager

FOR THE COMMANDER

MICHAEL G. MACNAUGHTON, COL, USAF
Deputy Commander Development & Acquisition

Please do not request copies of this report from the Human Systems Division. Copies may be obtained from DTIC. Address your request for additional copies to:

Defense Technical Information Center
Cameron Station
Alexandria VA 22301-6145

If your address has changed, if you wish to be removed from our mailing list, or if your organization no longer employs the addressee, please notify HSD/SORT, Brooks AFB TX 78235-5000, to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY 2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
4. PERFORMING ORGANIZATION REPORT NUMBER(S) Project 04515, Report 6624, NSBIT Task Order 0003	5. MONITORING ORGANIZATION REPORT NUMBER(S) HSD-TR-89-010-Vol.II
6a. NAME OF PERFORMING ORGANIZATION BBN Systems & Technologies Corporation 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Advanced Development Program Office HSD/YA-NSBIT
6c. ADDRESS (City, State, and ZIP Code) 21120 Vanowen Street Canoga Park, CA 91303	7b. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB OH 45433-6573
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Advanced Development Program Office 8b. OFFICE SYMBOL (If applicable) HSD/YA-NSBIT	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F33615-86-C-0530
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB OH 45433-6573	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. 63723F PROJECT NO. 3037 TASK NO 02 WORK UNIT ACCESSION NO. 01

11. TITLE (Include Security Classification)

Initial Development of an Assessment System for Aircraft Noise, Volume II: System Design Strategy

12. PERSONAL AUTHOR(S)

Fidell, Sanford; Reddingius, Nicolaas; Harris, Michael; and Kugler, B. Andrew

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM 2/12/87 TO 7/31/89		52

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	
12	07		Environmental impact assessment NEPA aircraft noise

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This is the second volume of a four volume report summarizing the development and current contents of a preliminary prototype version of an Assessment System for Aircraft Noise (ASAN). ASAN is a computer-based system intended to assist members of the United States Air Force (USAF) environmental planning community in addressing noise-related issues in developing environmental impact analysis documents, in compliance with USAF and other regulations, especially the National Environmental Policy Act (NEPA) of 1969.

This volume describes the development and capabilities of the preliminary prototype version of ASAN and recommends actions needed to develop a final prototype version.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Robert C. Kull, Captain, USAF	22b. TELEPHONE (Include Area Code) (513)255-3384 22c. OFFICE SYMBOL HSD/YA-NSBIT

DD Form 1473, JUN 86 Previous editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

Table of Contents

List of Figures

List of Tables

Acknowledgments

ASAN is a product of the labors of a number of people. We gratefully acknowledge the following contributors and their substantive contributions.

Mr. Gary Dworman assisted in porting, debugging and enhancing much of GRASS-derived graphics code from the UNIX/MASSCOMP environment to MS-DOS. Mr. Suyeo Tomooka implemented most of the user interaction screens. Ms. Linda Secrist wrote most of the noise effects calculation and report generation modules and assisted greatly in system integration. Ms. Laura Silvati digitized all of the cartographic information in the current version of ASAN. Mss. Barbara Nicholson, Michaela Heeb, and Elizabeth Lantos helped to design database schemata, develop NEPA checklist procedures and enter bibliographic information. Ms. Mara Feeney provided guidance on NEPA requirements. Mr. Joseph H. Marti assisted in the development of database entry and conversion code. Mr. Karl Pearsons was instrumental in conducting reviews of the human effects literature.

The animal effects literature database was designed and partially compiled by Dr. Frank Awbrey and Ms. Ann E. Bowles of the Hubbs Marine Research Center of Sea World. Mr. Doug Gladwin of the U.S. Fish and Wildlife Service was responsible for a large portion of the animal effects literature contained in the database. The structural effects database was compiled under the direction of Mr. Jerold M. Haber of NTS Engineering.

In addition, we are pleased to acknowledge the advice of Dr. William Galloway on noise-related issues; discussions with Mr. William Goran and his associates at the U.S. Army Construction Engineering Research Laboratory and with Mr. Richard C. Durfee of Oak Ridge National Laboratory about geodatabase systems; and discussions with Mr. Alton Chavis of TAC, and with Ms. Gloria Hagge and Mr. David A. Henney of SAC, about the content and structure of ASAN.

We also acknowledge the direction provided to this effort by Messrs. Lawrence S. Finegold and Preston Hall, and Lt. Col. Geral Long of the Noise and Sonic Boom Impact Technology (NSBIT) program Advanced Development Program Office (ADPO).

Executive Summary

This is the second volume of a four volume report summarizing the development and current contents of a preliminary prototype version of an Assessment System for Aircraft Noise (ASAN). ASAN is a computer-based system intended to assist members of the United States Air Force (USAF) environmental planning community in addressing noise-related issues in developing environmental impact analysis documents, in compliance with USAF and other regulations, especially the National Environmental Policy Act of 1969 (NEPA, 1969).

Volume I of this report is an Executive Summary. This Volume describes the development and capabilities of the preliminary prototype version of ASAN and recommends actions needed to develop a final prototype version. Volumes III and IV contain technical appendices and listings of the source code for the preliminary prototype version of ASAN.

1. INTRODUCTION

This report summarizes the initial development of a computer-based Assessment System for Aircraft Noise (ASAN) that is intended for use by members of the United States Air Force (USAF) environmental planning community in developing the noise-related portions of Environmental Impact Assessment (EIA) documents, especially for Military Operating Areas (MOAs) and Military Training Routes (MTRs). These documents (such as Environmental Impact Statements (EISs), Environmental Assessments (EAs), Findings of No Significant Impact (FONSIs), etc.) are required by various USAF congressional, and Environmental Protection Agency (EPA) regulations, including the National Environmental Policy Act (NEPA).

The product of this initial development effort is a preliminary prototype system that is useful primarily as a proof-of-concept system for demonstration use. The present plan for ASAN development is for the work on the preliminary prototype version to be followed by development of a final prototype, then by a period of trial use and evaluation, and finally by production of a formally released version of ASAN. The rationale for development of ASAN, as well as preliminary and detailed system specifications, have been described in separate documents (Fidell and Harris, 1987; Harris and Fidell, 1987; Fidell, Harris and Reddingius, 1988) which are summarized in part in this volume.

Effort during the first 6 months of this project concentrated on system definition. The Statement of Work for this project assigned to the contractor the responsibility for analyzing the needs of USAF environmental planners, and for recommending and then designing a system that could provide the required tools. As described by Fidell, Harris, and Reddingius (1988), the primary user for whom ASAN is intended is a USAF officer or civilian serving in a major command or airbase-level environmental planning office. This end user is expected to have only rudimentary computer knowledge, limited access to a computer larger than a desktop microcomputer, and only a basic understanding of environmental acoustics. Although ASAN contains a number of features which may be exploited by more sophisticated users, the user interface and general mode of operation of the software are designed for inexperienced users.

The following were among the principles adopted in designing ASAN:

- Software should facilitate the basic aspects (e.g., text, graphics, and document handling) of the environmental planner's job.
- Software tools should not be limited to those which implement the way that environmental planners currently perform their jobs, but instead should provide environmental planners with tools that make it possible to work more effectively and productively.
- Software should be modular (producible and operable in parts) and expandable as

time and budget permit, so that it can potentially provide assistance in all aspects of the environmental planner's job.

- Software should be organized in such a manner that it can take maximal advantage of existing programs and databases. It should not be developed from scratch if it can obtained by modifying existing code, and it should not be developed at all if the desired capability can be purchased at reasonable expense.
- As much of the software as possible should be executable on desktop microcomputers readily available to environmental planners.
- The software system should permit the burden of updating and maintaining the databases on which it relies to be shared in an efficient manner between a central responsible organization and local environmental planning personnel.

Chapter 2 provides background information about design recommendations and system specifications for ASAN. Chapter 3 summarizes implementation decisions for the preliminary prototype version. These initial chapters paraphrase text contained in Fidell and Harris (1987), Harris and Fidell (1987), and Fidell, Harris, and Reddingius (1988). Chapter 4 describes the strategy devised for developing the code needed for the proof-of-concept demonstration. Chapter 5 summarizes the capabilities of the preliminary prototype code. Chapter 6 discusses the contents of the databases included in the preliminary prototype. Chapter 7 suggests directions for the development of the final prototype system (an alpha-test version of ASAN). The appendices in Volumes III and IV contain details of the source code for the preliminary prototype system.

2. BACKGROUND

2.1 Current Air Force Procedures for Performing Environmental Impact Assessments

The overall goal of the current effort is to increase the credibility and completeness of USAF EIAs by enhancing the efficiency and productivity of those who must prepare such materials. A system that can do this must assist environmental planners with both their procedural (i.e., document handling) and substantive (i.e., informational and decision making) needs. Design of such a software system is complicated by organizational and other difficulties, as discussed in this section.

One organizational difficulty encountered in designing environmental planning aid software is the fragmentation of responsibility within the USAF for noise-related EIS. The major commands differ in their approaches and resources committed to environmental planning. Half a dozen (or more) organizations within the USAF have related but only loosely coordinated interests in noise-related matters; contractor-operated facilities and computational resources are scattered geographically and organizationally; and multiple organizations throughout the Department of Defense (DOD) and in other federal and local agencies have overlapping capability, interests, and development plans.

Other factors also affect the degree of cooperation and acceptance that is likely to be accorded to NSBIT-sponsored environmental planning aids. These factors include most obviously the cost and complexity of the software, its accessibility by interested parties, and its reliability, ease of use, and utility to end-users of varying skill levels and interests.

Perhaps the most basic prerequisite for acceptance of environmental planning aid software, however, is its usefulness. A system that provides simple and effective solutions to common, time consuming, and frustrating problems is more likely to be greeted enthusiastically than one which provides only a small increment in capability for dealing with minor aspects of the EIA process. The wishes and needs of the end-users of the desired environmental planning aid software system must therefore be understood and seriously considered in the present design effort. It is for this reason that conversations and visits were undertaken early in the current effort with a variety of USAF and other personnel involved in EIAs.

It is helpful to explicitly identify the anticipated users of the desired environmental planning aid software. The two types of end-users who could derive the most immediate and useful assistance from such software are (1) base-level personnel at Tactical Air Command

(TAC) facilities, and (2) headquarters staff at both TAC and Strategic Air Command (SAC). Additional users may include personnel of other USAF major commands. Some incidental use of the software (e.g., for development, demonstration, or maintenance purposes) may also be made by USAF contractors, as well as NSBIT project office staff. In the long run, it is possible that other DOD agencies and other government organizations may also find uses for environmental planning aid software.

Although various parties have different perspectives on the utility of environmental planning aid software, a number of common concerns are apparent as well. In general, DOD personnel involved in performing environmental impact assessments are forced to do too much with too little. Office space is at a premium; support personnel are scarce; library, computational, documentary, graphic, and other material resources are meager; the press of deadlines is continuous; and the overall workload is heavy. Few have the time to do as good a job as they would like, so that compliance with the requirements of NEPA is sometimes less than ideal.

2.2 The Nature of the Environmental Planning Process

The USAF must carry out EIAs at local (base) and central (headquarters) levels in each major command to conduct its flight activities in as environmentally acceptable a manner as possible. Local planners at USAF flying installations are often responsible for dealing with environmental matters associated with flight and other operations.

Although the major commands differ in their degree of centralization of environmental planning, the local environmental planner is often a junior officer charged with other responsibilities as well. The bulk of the noise-related portion of the local planner's job is to prepare assessments of potential changes in environmental impacts associated with planned changes in flight operations. Consequently, the local planner (especially in TAC) must keep in touch with local civil authorities to maintain a current understanding of community developments.

For example, a local planner may attend zoning commission hearings, meet with local government officials and citizens' groups, and respond to complaints and other community concerns about USAF operations. The local planner commonly has a civil engineering background, and is rotated in and out of an environmental planning job over a period of 1 or 2 years. The planner's replacement must typically start all over to regain the benefits of his predecessor's experience. The local planner generally has little technical background in acoustics (much less in the effects of noise exposure on people, animals, and structures), a

limited amount of noise-related training, and a short course in NEPA requirements. Much of his training is on-the-job.

Environmental planners in TAC are more often concerned with noise produced by flight operations in and around MOAs, while those in SAC are more often concerned with noise produced by flights along MTRs. Planners in both commands are likely to be involved in and familiar with USAF Air Installation Compatible Use Zone (AICUZ) procedures for dealing with approach and departure noise in the vicinity of air bases.

The base commander has broad discretionary authority over aircraft movements, so that an environmental planner may not immediately learn about minor actions which are not perceived as having any adverse environmental consequences. The base operations staff generally informs the environmental planner of the need for an environmental impact assessment only if there is reason to believe that a proposed change in flight activities might provoke public opposition.

When notified, the environmental planner attempts to define the scope of the proposed action in terms of numbers of sorties, types of aircraft and mission involved, route changes, and so forth. Many such impact assessments are little more than cut-and-paste adaptations of previously completed documents, and are produced principally so that the appropriate paperwork will be on file if a challenge ever arises to a particular action. Such challenges are relatively infrequent, but can come to the attention of headquarters staff with little warning, as a consequence of local political opposition, litigation, or a congressional inquiry.

It has not been uncommon in the past to find that documentation for a FONSI is incomplete or inaccurate, although the recent trend has been toward fuller compliance. Operational information is sometimes skimpy, maps may be unclear, or the thoroughness of the EIA leading to the FONSI may leave something to be desired.

Such flaws in the routine aspects of an environmental planner's work detract from the overall credibility of USAF compliance with NEPA requirements. As local government planning staffs become increasingly sophisticated, their demands for accurate and credible information about USAF actions which affect the environment are increasing. The current system of environmental planning is not able to meet these increasing local demands for standardized reports of higher quality.

Headquarters personnel involved in environmental planning have much more training and experience in environmental acoustics than local planners, and tend to remain in the same positions for longer periods of time. A field grade officer usually heads the environmental office at headquarters level, and is assisted by other senior military and civilian personnel. Headquarters staff organize training materials for local planners, provide some forms of assistance to them, and review their efforts. They may also assume complete responsibility for EIAs of larger scale or potentially controversial actions.

Headquarters staff has better access to USAF-wide technical and computing resources than local planners, and may also obtain contractual assistance from environmental planning organizations for major projects. They do not, however, have staffs adequate to scrutinize every document produced by local planners, nor to assist them in keeping their files up to date.

Personnel involved in environmental planning at headquarters level must often operate in a reactive, event-driven mode. They typically respond to requests for EIAs produced by route planners, or to congressional inquiries, or to other forms of public comment. They do not often have the time or resources to initiate independent activities that are not related to specific problems, nor to interact extensively with other USAF personnel in the early stages of operational or route planning.

2.3 Materials and Information Manipulated by Environmental Planners

Since NEPA defines environmental impacts very broadly, USAF environmental planners must deal with many different informational documents and materials in the course of their work. In the case of noise exposure effects, the information of interest concerns effects of noise on humans, animals and structures.

2.3.1 Graphic Materials

The fundamental materials which define an environmental planner's job are graphic in nature, including maps of many kinds and various types of imagery and remotely sensed data. These materials are generally awkward to manage (acquire, store, retrieve, manipulate, update, examine, modify, compare, and combine). Environmental planners work with maps in every facet of their jobs: laying out MTRs and MOAs, estimating noise exposure from MTRs and MOAs, locating sensitive land use areas, examining alternative courses of action, predicting impacts, and so forth.

Current access to maps and charts might best be described as catch-as-catch-can. There is no central repository of off-base maps which environmental planners can easily call upon for their various needs. There is also little or no capability for producing or modifying three dimensional graphics, which can be especially important for displaying airspace management and flight profile information. By the time graphic materials appear in published documents, they are often photocopies of photocopies, only marginally legible.

Environmental planners in TAC are generally concerned with maps, imagery and remotely

sensed data within a 500 mile radius of their facilities, a land area of almost 800,000 square miles. Environmental planners in SAC are often concerned with graphic materials describing land areas adjacent to MTRs at distances of hundreds if not thousands of miles from SAC facilities.

2.3.2 Textual Materials

Textual materials are next in importance to environmental planners. The texts they must manipulate (acquire, create, modify, search, cite, and abstract) include documents of several sorts. These documents can be voluminous, not to mention difficult to access, interpret, and keep current. Assembling these materials into public documents is a difficult task. It is not uncommon to find DOD-produced environmental documents with pages upside-down or numbered out of sequence, or with missing text or mislabeled tables, figures, etc. While such minor flaws may not affect the technical quality of the document, they detract from the overall credibility of EIA documents.

Many of the texts which environmental planners must edit concern effects of aircraft noise exposure on humans, animals, and structures. Some of this text is boilerplate, i.e., standard wording that is reused verbatim in different documents. Often, however, boilerplate must be edited to adapt it to immediate purposes. Several problems are encountered in modifying noise effects text. First, not all environmental planners have the technical training or familiarity with the noise effects literature needed to make substantive changes to the text. Second, the text itself is not generally organized in a fashion that is readily tailored to differing purposes, nor is it generally stored in a form convenient for manipulation. Third, many planners have no easy way to ascertain that technical information in the text is current, nor to update the text from recent publications in the technical literature.

Thus, texts which need any more than minor reworking may be omitted from EIAs, even when it is known that their inclusion would strengthen the documents in which they belong.

2.3.3 Calculational Models

Environmental planners must also exercise a variety of calculational models for purposes of estimating noise exposure and impacts of aircraft operations. These models include aircraft noise emission models, acoustic propagation models, structural damage models, point and area exposure estimation models, population impact models, and annoyance prediction models. Planners also require access to historical data on operations, exposure, and impact analyses to compare against current or proposed conditions.

2.3.4 Nonacoustic Materials

Beside the just mentioned (that is, noise-related) graphics, text, and models, environmental planners also deal with similar materials relevant to aircraft emissions and impacts other than noise, notably air quality and economic impacts.

2.3.5 End Products

Individual environmental planners may have to produce dozens of EIAs per year to comply with NEPA requirements. These assessments can lead either to a FONSI or a Categorical Exclusion (CATEX) in the case of minor operational changes for existing MTRs and MOAs, or to a full-scale EIS in the case of a major federal action. Both types of documents are supposed to be preceded by a full description of the proposed action, including complete and accurate statements of operational information.

The elapsed time between the start of planning of an MTR and issuance of a FONSI can be nearly 2 years. A MOA can take even longer to plan, environmentally assess, and put into operation. The route planning effort itself requires consideration of many nonenvironmental issues and coordination with various government agencies. It can take 15 months or more to complete. Another 6 months may be required to complete an EIA of a proposed route.

A locally produced Environmental Assessment Certificate for a FONSI is often a 10 to 20 page document that includes both text and graphics. Some FONSIs, especially for long MTRs that entail low level flight by large bombers over several large states, can be much longer. These documents are commonly produced by example; that is, with reference to a previously filed certificate that is modified to show new routes and describe new impacts. Base-level planners who prepare such certificates may have little contact with the technical literature on noise effects, and few facilities for keeping abreast of and evaluating new information about noise impacts. Hundreds of these FONSIs are forwarded to headquarters staff each year for review and improvement (as time permits).

An EIS is a much more massive document that must be prepared to describe major operational changes (for example, in use of existing MOAs or in justification of a new MTR), or as part of the description and justification of a new MOA. Preparation of an EIS is a time-consuming process that includes lengthy periods of public comment and reply. All materials produced in the course of preparing an EIS must be able to withstand detailed scrutiny in the not unlikely event that political or legal attention is drawn to the proposed USAF action. A complete record of decision that can serve as an audit trail for justifying consideration and rejection of alternate actions is also required by NEPA.

2.4 Computer Based Environmental Planning Resources

United States Air Force environmental planners presently perform most of their functions with only minimal computer-based assistance. They have little local capability for dealing with graphic-oriented materials; little text handling capability beyond word processing; and until recently, only remote, batch mode capability for exercising noise exposure models. However, most environmental planners, even at base level, do have access to IBM-equivalent personal computers. Many lack the training and communication capability to make efficient use of more sophisticated computing resources.

Various organizational units throughout DOD have developed computer-based tools that can facilitate some parts of the environmental planner's job, some of which are described later. Because most of these resources are not integrated into a single easily accessed system, not all are widely used by USAF environmental planners.

2.5 General Description of ASAN Operation

The primary mode of operation of ASAN is intended to be interactive and iterative in real time. Since an environmental planner is unlikely to have at hand all of the information or all of the time necessary to complete an EIA in a single computing session, several provisions are made in the organization of the system to permit users to enter and manipulate even small amounts of information efficiently. The software is also organized to permit environmental planners to consider several alternative proposed actions at a time, either as variants of the same action or as independent projects.

Figure 2-1 is a block diagram of the functional capability of ASAN from the user's point of view. The first phase of the EIA process is inevitably problem definition. Users must supply the system with at least fragmentary information about the noise sources and geographic areas of concern. As more detailed information about mission requirements, land uses, and the like becomes available, users continue to interact with the software modules that accept geographically oriented problem definition data. This iterative problem definition phase is expected to be convenient enough that users will eventually rely on computer based methods for data entry and retrieval to organize their routine handling of documents used in the EIA process.

End users will acquire some of the relevant data, such as terrain maps, land use maps, and aircraft noise emission levels and contours, from centralized sources. Other data, such as proposed routes, locations of residential areas, and contact lists, will be obtained and entered

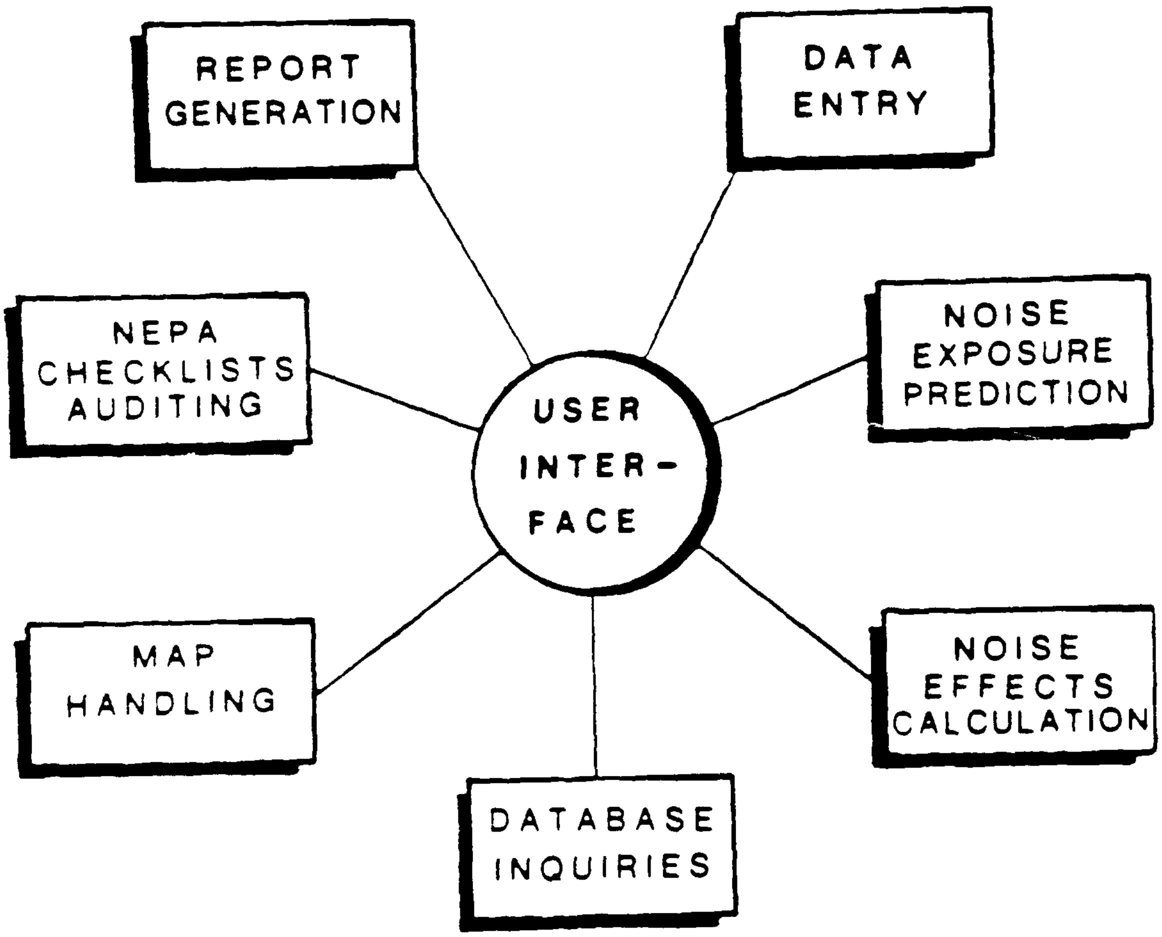

Figure 2-1: ASAN from the User's Perspective.

locally. The product of this phase is a geographically organized composite data structure, with many superimposed "layers" of information.

The next step in conducting an EIA is analytic. The initial part of the analysis generates estimates of the noise exposure created by flight activity for specified points or areas. Exposure estimation is also expected to be an iterative process in which users may compare in increasing detail the relative contributions of a variety of noise sources to total exposure. All such estimates are stored in geodatabases for convenience of manipulation and interim display.

The final portion of the analytic process is estimation of effects. A number of analytic procedures, implemented as independent software modules, operate on the layered composite data structure. The ultimate product of this phase is a group of global values and data blocks available to the next (report generation) phase.

In the report generation phase, the collected data and analysis results are combined either automatically or under user control to produce text and graphic material for interim and final reports. The data and displays produced during the analysis phase are used to generate documents containing both prose and figures.

3. IMPLEMENTATION DECISIONS

3.1 Selection of Host Computer

The portion of the preliminary prototype version of ASAN made available to end users is intended to be hosted at least in the short term on the Zenith Z-248 personal computer under the MS-DOS operating system. The reasons for this selection are:

- These machines are of reliable and proven design, and have been purchased in large quantities by the USAF. Since they are already part of the USAF inventory, they are likely to be available to environmental planners at the base level without special procurement efforts.
- They are sufficiently fast and powerful to support the required tasks locally, in a stand-alone environment, without extensive interaction with centralized facilities.
- They provide fully compatible upgrade paths from the Intel 80286 to the Intel 80386 processor, and from the present single-user, single-task operating system to a concurrent (background/foreground) operating system which is expected to be announced in 1988. These hardware and software upgrade paths minimize the effort required to accommodate future advances in personal computer technology which the USAF is likely to adopt by the time ASAN is ready for distribution.
- They are already the host computers for much of the existing (commercially available and previously developed) software to be included in ASAN.
- They represent a significant market force, so that many applications programs, including a number developed under USAF sponsorship, are available in compatible versions.
- The additional peripheral cards and devices necessary to support ASAN are available relatively inexpensively to fit the machines' standardized electronic connection bus.
- Since the machines and their MS-DOS operating systems are in common use in USAF management, training and documentation problems are small with respect to those that would attend selection of a different computing environment.
- The installed base of this and other IBM PC/AT compatible computers, both within the armed services and in the commercial arena is very large, assuring that their components and design philosophy will be supported well into the future.

3.2 Consideration of an Alternative Host Computer System

An alternative host computer system, described by the Federal Systems Division of Wang Laboratories, Inc., was also considered but found to be inappropriate for present purposes. This system, provided to the USAF under Contract F19630-86-D-0001, consists of two groups of hardware products.

1. The visual signaling (VS) computers, roughly equivalent in processing power to Digital Equipment Corporation's VAX products, are more powerful and costly than the EIA task requires. These machines are intended for multi-user office automation tasks. The Wang VS configuration guide does not list most of the hardware needed for ASAN, such as high resolution graphics, touch screens, and pointing devices. Since the multiplicity of suppliers and economies of scale which operate to provide high quality, low cost peripheral devices for IBM PC/AT compatible systems do not apply to VS computers, the availability of these devices is likely to be poor and their cost high.

   Furthermore, the VS computers use a proprietary "virtual storage operating system." It is unlikely that the third party software to be utilized in ASAN has been written for, or can be easily translated to, this operating system. Development of such software would add significantly to the cost and elapsed time necessary to complete the project.

   There is a Unix-equivalent operating system available for the VS computers. Its use would enable the use of a wider range of third party software; however, this operating system is not available to the USAF under Contract F19630-86-D-0001 and hence would be difficult and costly to acquire and maintain. In addition, the user would be required to learn an entirely new operating environment.

2. The second Wang product group, "Programmable Workstations" (also referred to as "Wang PCs"), is built around IBM PC-equivalent machines using the Intel 8086 microprocessor. Although IBM-compatible third party hardware and software is explicitly supported, an 8086-based machine would not be sufficiently powerful or flexible to perform the necessary operations with acceptable speed.

3.3 Hardware Subsystems for ASAN Preliminary Prototype Version

The hardware subsystems described later must be added to the basic host computer to provide the full hardware capability needed for ASAN. Not all of this hardware capability is needed for the prototype version of ASAN.

3.4 Mass Storage Subsystem

The basic Z-248 host system will include a nonremovable hard disk for system software storage. An additional mass storage device will be necessary to provide adequate storage space and reasonable access speeds for the databases which constitute the heart of ASAN. This additional storage will be removable to simplify updating and to allow the environmental planner to switch among several ongoing tasks. It is expected that the additional storage capability will be provided by removable magnetic disks in the demonstration system.

Digital optical disks are probably more suitable for future versions of ASAN. These devices can store approximately 500 megabytes in a small physical space, are less expensive than comparable quantities of magnetic storage, and are fairly rugged.

3.5 Graphics Display Subsystem

An add-on color graphics card and color graphics display monitor is necessary to permit display of graphic information. The card, an Imagraph IP1076-10-N-A-1.2-PC, provides resolution of 1,024 by 768 pixels, displays as many as 8 bit planes, and has the ability to conveniently accept input from the graphics input subsystems described later.

The graphics display device is a 48.26 cm (19 in.) (diagonal measurement) monitor distinct from the operator's console monitor that displays flicker-free full color images.

3.6 Graphics Input Subsystems

These add-on graphics input devices are needed to support ASAN's graphics manipulation capabilities.

3.6.1 Touch Sensitive Screen

A touch sensitive screen mounted directly on the graphics display monitor allows the environmental planner to indicate points or areas of interest and select displayed features rapidly and unambiguously.

3.6.2 Graphics Tablet

A graphics tablet permits entry of local maps and other data by direct tracing.

3.6.3 Console Pointing Device

The basic Z-248 personal computer must include a console display subsystem consisting of a controller card (functionally equivalent to the IBM Enhanced Graphics Adapter) and an associated display monitor (NEC Multisync or equivalent). Users will view ASAN control screens and text on this monitor.

A pointing device for the console display (Microsoft Bus Mouse or equivalent) is needed to permit efficient interaction with the screen-oriented user interface.

3.6.4 Digitizing Camera

Although it is not necessary for the demonstration prototype, a digitizing camera may be included in future versions of the system. This device would allow photographic images, maps and other graphic data to be entered into ASAN databases rapidly and directly. An administrative decision based on organizational and cost issues will be needed to determine whether the USAF would prefer to centralize this capability or provide it to end users.

3.7 Hardcopy Subsystems

Capabilities for producing rough and final versions of text and graphics output products are required. The ability to produce final versions of text and graphics of sufficiently high quality for publication is likely to be too expensive to provide to every end user. Centralized reproduction facilities might therefore be needed to accommodate final publication needs.

However, every field installation of the system requires the ability to produce rough drafts and proof copies during the assessment process. A color thermal printer is recommended for this purpose.

3.8 Streaming Tape Subsystem

A streaming tape controller card and drive assembly are needed to provide efficient and reliable mass storage backup capability. This subsystem must be able to:

- operate according to straightforward and understandable control commands;
- fit all the files on the mass storage device onto a single tape cartridge; and
- back up and restore all files on the mass storage device, or just those files modified since the last backup was performed, or selected groups of files, or individual files.

Furthermore, the tape backup system must be supplied with vendor developed and supported software, providing all necessary functions in the form of C language callable subroutines, and must use standard tape cartridges, preferably in a standardized recording format.

3.9 Telecommunication Issues

The need for telecommunication is a basic issue that affects many aspects of the design of the current software system. It is useful to distinguish various types of communication that are adequate for various purposes.

The most basic approach to telecommunication among geographically distributed computers and users is aptly described by the phrase "diskette-net." Information in a diskette-net is interchanged among users and hosts by mailing distribution materials (floppy disks, digital optical disks, magnetic tape cassettes, etc.) by conventional mail. The diskette-net approach to telecommunications is slow, but it can be effective as an inexpensive solution to many communication needs. Its operation is intuitively obvious to users of all skill levels (it requires no skills beyond those required to mail parcels), and provides a straightforward solution to configuration control of databases and software releases: "Mail back last month's media when you get this month's." This method of communication can also provide unsophisticated users with the additional benefit of automatic backup protection.

Diskette-net can also be a method of choice for distributing large amounts of information at reasonable cost. Widely available and inexpensive floppy disks can store more than a megabyte of data. Order of magnitude improvements in storage density for magnetic media (10 megabyte 5.25" floppies, for example) have been announced, and should be commercially available in the near term. Relatively inexpensive WORM (Write Once, Read Many) digital optical disks, based on the familiar compact disk technology for consumer high fidelity, have been available to microcomputer users for more than a year.

The next level of telecommunication capability is dumb terminal/ dial-up access via common carrier to a remote host computer. Such capability is now widely available only at transmission speeds of 1,200 baud or lower, although 2,400 baud communication is gradually becoming more commonplace. A screenful of text (say, 24 lines of 80 columns each) requires about a minute and a half to transmit at 1,200 baud. A printed, single spaced page of text (say, 60 lines of 80 columns), takes about 4 minutes. A 20 page document would therefore require over an hour to transmit at this rate.

Urgency is the only real reason to prefer this form of telecommunication over diskette-net. It is most appropriate for short, high importance message traffic (e.g., electronic mail), because a dumb terminal generally has no convenient way to capture, preserve, or edit keystrokes, nor to access text prepared by stand-alone word processors.

The disadvantages of the dumb terminal/common carrier dial-up method of communicating are not to be underestimated. Dial-up telephone connections can be expensive, not to mention unreliable over long periods of time and long distances. The cost of the required hardware--terminal plus modems at each end--can be nearly as great as for the next option discussed. There are also potential hardware limitations on the number of access ports that a host computer can support which limit the number of concurrent users of this form of communication.

The next most sophisticated form of telecommunication is a personal computer/dial-up access arrangement. The end user of such a system can prepare text off-line, then use local communication software to send this text to a remote computer. This arrangement also permits a user to capture text from a remote host for local use. It is subject, however, to all of the previously noted disadvantages (cost, reliability, quantity of information), and assumes a fair degree of user sophistication. To make good use of such a telecommunication capability, a user must be at least minimally familiar with the operating system, editor, and communication software capabilities of the personal computer, as well as the operating system, editor, and mailer of the remote host computer. A novice attempting to acquire the skill levels required to make effective use of this form of telecommunication could easily need several weeks of training, practice, and diligent effort, not to mention considerable patience and a high tolerance for frustration.

Most, but not all, of the unreliability of common carrier access to a remote computer host can be eliminated if access is provided by special purpose digital communication networks. A variety of private, commercial, and government-owned networks are available for such purposes. The costs of network access and use for different classes of users can vary widely, depending on how they are allocated to users and their organizations. As a rule, however, the increased reliability of such telecommunication is not inexpensive. Furthermore, the increase in reliability is not necessarily accompanied by an increase in speed of communication. Network communication also requires additional skills of users, who must learn yet another set of protocols for dealing with network access.

Physical exchange of media (diskette-net) appears adequate for all purposes of the prototype versions of ASAN.

4. SOFTWARE DEVELOPMENT STRATEGY

The entire schedule for coding the preliminary prototype version of ASAN was compressed into less than 6 months. Even though the major goal of the development of a preliminary prototype version of ASAN was to support a proof-of-concept demonstration, an important secondary goal was to avoid creation of "throw-away" code that could not be reused in subsequent phases of ASAN development. The decision was therefore made to preserve independence of control logic, application code, and data to the greatest degree possible. This philosophy was adopted to maximize the eventual transferability of ASAN to alternative hardware systems.

Fundamental architectural decisions (notably identification of the main software building blocks) thus had to be made very early in the development cycle. As shown in Figure 4-1, these major building blocks included the user interface and the text and graphics database management packages.

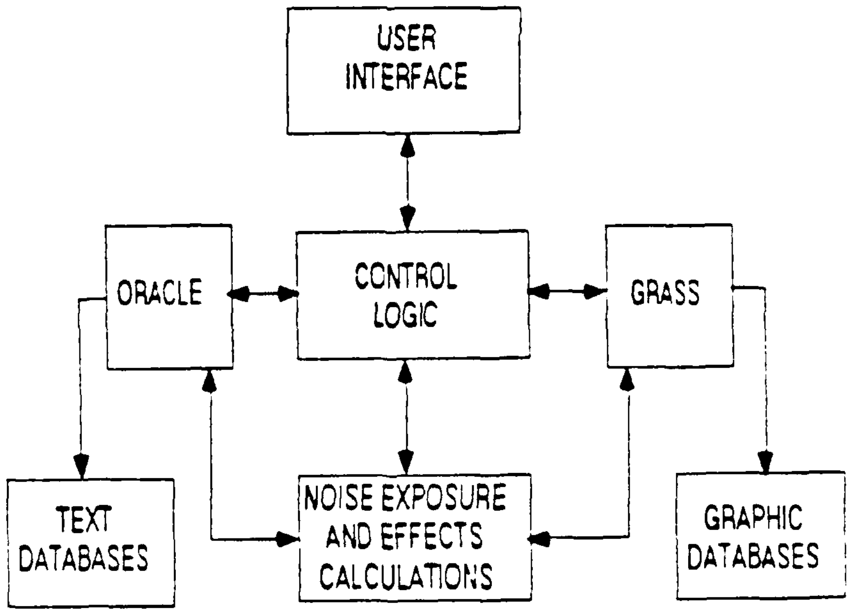

Figure 4-1: Major Building Blocks of ASAN Software.

The following subsections describe the approach used to create the executable code needed for a working system from the components shown in Figure 4-1. Creation of the databases upon which ASAN operates is described in Chapter 6. All high level code in ASAN is written in the C programming language to take advantage of the efficiencies of modem structured programming techniques, software libraries, compilation aids, and related software development tools.

4.1 User Interface

From the user's perspective, the look and feel of ASAN are determined by the user interface. This section describes the bases for selecting a screen-oriented interactive user interface for ASAN.

The compressed software development schedule for this effort required early adoption of a flexible user interface that permitted creation of a great deal of functionality in a short period of time. The user interface selected for ASAN is a BBN-proprietary package known as U that produces viewable screens composed of "pickable" and "nonpickable" elements (those on which the cursor may and may not rest, respectively). Pickable elements permit users to supply information or command program actions. Nonpickable elements display information about the current state of the program. Both sorts of elements may be combined in windows that may be displayed on any screen, and in pop-up windows that can alter the underlying appearance of windows contingent upon user interaction. Appendix A of Volume III documents the capabilities of this user interface. Since U was originally developed in the UNIX environment, its adoption required porting it to MS-DOS for use with ASAN. BBN provided the U package for use in ASAN at no cost to the government.

Selection of U as the user interface provided several substantial benefits both to end users and to the software development effort. For the unsophisticated end user, U provides a consistent and simple screen-oriented interface for all interactions with ASAN. It shields the user from the complexities of ASAN's underlying software packages (i.e., ORACLE and GRASS, described later) and analytic code (e.g., noise exposure and noise effects calculations). Other desirable features of this user interface for the intended end users and manner of use of ASAN include context-sensitive, on-line help and a nonhierarchical mode of operation.

The principal benefits of U for the software development effort included increased modularization of ASAN's functional capabilities by isolating all interactive input/output operations in an interface package rather than dispersing it through the application code: straightforward and consistent conventions for creation of control logic; direct C-language

interfacing to other application code; independence of external program appearance from internal functioning; and rapid creation of viewable screens.

Since user interface screens vary in appearance interactively, it is not possible to fully illustrate them in this report. Hardcopies of most of the major aspects of the user interface screens (e.g., with and without pop-up windows, data entries, selected actions, etc.) may be found in Appendix B of Volume III. Listings of help files and text blocks associated with these screens may be found in Appendix C of Volume III. The screen description format files that create ASAN's viewable screens may be found in Appendix A of Volume IV of this report.

4.2 Relational Database Management

The second major building block on which ASAN was developed was a relational database management system. The rationale and selection criteria for this software are discussed in this section.

4.2.1 Rationale for Data Management in ASAN

In principle, a relational database management system can completely decouple an application program from the physical organization of the data on which it operates. This makes it possible for data stored in one place to be accessible to all application programs, while greatly simplifying the maintenance of the data. A relational database management system also makes it possible to perform *ad hoc* (i.e., unanticipated) queries with relative ease.

Commercially available database management systems offer a set of tools that allow considerable separation between application code and data management. For example, changes in the physical structure of the data can often be implemented outside application code, or may require no more than simple recompilation without any changes to application code.

The separation of application and data has other important consequences. One of the key problems in algorithm-driven data processing is that the same data may be replicated in different files. Keeping these files synchronized is a major difficulty. Formal technologies exist to develop databases that contain neither duplicate information nor empty records. Such normalized databases make the most effective use of physical storage media.

After the initial effort of designing and implementing a normalized database, subsequent code development and maintenance is less costly in both time and money. Also, it is possible to perform *ad hoc* queries on a normalized database with relative ease. For information retrieval

and decision support systems such as ASAN that must function in an unstructured evaluation and decision making environment, these capabilities are essential.

Database management systems may be characterized as predominantly high-volume data capture (transaction-oriented) systems, or predominantly analytic (management information and decision support) systems. The former support a highly structured work environment, requiring little or no user judgment; the latter support tasks that have relatively little inherent structure.

ASAN, intended to facilitate the conduct of environmental impact analyses, is in this latter category. The specific actions to be taken for an individual assessment vary widely from case to case. The data on which assessments must be based come from a variety of sources, and may need considerable reworking before they are in a form suitable for analysis. Furthermore, the state of the art in noise impact prediction is evolving as more is learned about the effects of noise on people, animals and structures. Clearly, this calls for a highly modular system design, both with respect to the design of the application software and of the data structures.

ASAN insulates the primary user from direct creation and manipulation of most of the databases it contains. This action reduces the end user's needs for training and computational skills, while encouraging concentration of effort on generation of environmental impact analysis work products. This design goal was implemented in the preliminary prototype version of ASAN by interposing a layer of control software between the database management software and the user.

More sophisticated users of ASAN are provided with direct access to ASAN's database through the database manager's implementation of the Structured Query Language (SQL).

4.2.2 Selection Criteria for Text and Numeric Database Managers

The primary criteria for selecting database management software for incorporation into ASAN were as follows:

- Availability of a version that can execute in IBM PC/AT MS-DOS and Zenith Z-DOS operating environments.
- Ability to rapidly and efficiently locate records.
- Ability to accommodate variable length records.
- Minimal consumption of system resources such as volatile and mass storage.
- Absence of gratuitous capability, especially that which increases software or database size or decreases system performance.
- Ability to operate while co-resident in main memory with other software, taking advantage of whatever remaining memory is available.

- Imposition of minimal data input and storage constraints such as arbitrary limits on number of bytes in a field, number of fields in a record, or number of records in a database.
- Implementation of all capabilities as a library of functions callable from external C language programs.

A less stringent set of criteria was used to select database management software for database creation. Since the initial data entry did not require the speed or sophistication of the retrieval system, a less capable database management system sufficed.

4.2.3 Selected Database Management Systems

ORACLE, a fully-featured relational database management system, was selected for incorporation into ASAN. It is a complete implementation of SQL, and also includes a report generator, a text formatter, on-line help, and extensive auditing and file security subsystems, as well as spreadsheet, business graphics, and forms management modules.

A version of ORACLE compatible with the Zenith Z-248 computer was released shortly before development of ASAN began. A unique feature of ORACLE's implementation for the Z-248 is that it operates as a separate process in the Intel 80286 processor's protected mode; that is, it uses memory above 640 kbytes which is not used for applications running under MS-DOS. Operation of the database management system in protected mode therefore leaves most of the machine available for ASAN's control logic, geodata management, and computational code.

dBASE III+, a popular database management system derived from software written for 8 bit microcomputers, was selected for initial data entry purposes for most of ASAN's databases. Its principal advantages were that it was readily available and well understood by those preparing the databases, and sufficiently powerful to accommodate ASAN database schemata. Information entered through dBASE III+ was converted to ASCII files and loaded by ORACLE's ODL utility.

4.3 Geodatabase Management Software

The geodatabase management system selected as the basis for the ASAN geodata software is a public domain software package written in the C language at the U.S. Army Construction Engineering Research Laboratory (CERL). Version 2.0 of this Geographical Resources Analysis Support System (GRASS) was released shortly before development of ASAN began.[1]

GRASS is a modern, interactive grid cell (raster) based system with a capacity for rapid production of extensive map comparisons and overlays. Among the attractive features of GRASS for present purposes are that it performs its operations somewhat faster than other geoinformation systems it accepts digital geodata in a variety of formats; and it can resample data to change scales efficiently.

GRASS is preferred to other geodatabase systems for the following reasons among others:

- GRASS geodata formats are reasonable, well defined, very efficient, and proven in actual use. GRASS can store data in compressed forms to reduce mass storage requirements for geodatabases.
- GRASS produces well-composed and formatted graphic output, in color, on a variety of display and hardcopy devices. Its graphic output functionality is highly modular and largely device independent. Output devices can be rearranged and substituted, and graphic output commands can be redirected to mass storage for later printing or display.
- GRASS is structured as a library of independent subprograms, so that unnecessary functionality can be easily omitted.
- The original software development group is intact and actively working on enhancements to the system. Code development is proceeding in a systematic and rigorous fashion, in a modern development environment. A single agency, CERL, has complete authority and responsibility for the development of GRASS, and is providing excellent documentation and support for the package.
- Source code for GRASS is available for the minimal cost of copying distribution media.

As developed and distributed by CERL, GRASS Version 2.0 was intended for and supported on Masscomp and Sun computer systems under the UNIX operating environment. The ASAN prototype development plan included converting the CERL GRASS software to run on the Z-248 ASAN host computer system under the MS-DOS operating environment; that task was expected to involve conversion of the lower levels of existing GRASS code modules.

---

[1]Version 3.0 of GRASS was released shortly after completion of the initial development work described in this report.

Once the task of porting GRASS to the MS-DOS operating environment was under way, it became clear that the production of an efficient MS-DOS version would involve more extensive code revision than had been expected. Much of the original GRASS code was found to depend on particular hardware and operating system environments. This finding occasioned the rewriting of GRASS C code in a more generic form. Furthermore, certain features of CERL GRASS proved to be superfluous in the context of a software system such as ASAN that incorporates a powerful database manager. Likewise, some features of CERL GRASS simply could not be accommodated in an MS-DOS environment. For example, the single user MS-DOS version does not require the capability of UNIX GRASS for providing multiple users access to the same files. Certain redesigns (and in some cases complete reimplementations) of some of GRASS modules were therefore undertaken for all of the reasons just mentioned.

Thus, while the external appearance and functionality of ASAN GRASS are very similar to CERL GRASS, the two implementations differ internally. Nevertheless, the geodata file formats used in ASAN GRASS are exactly those used in Masscomp/Sun GRASS, and geodata files created in either environment can be processed and displayed in both. Any further porting of GRASS modules to provide ASAN with more of the functionality of CERL GRASS (e.g., image processing algorithms) should require considerably less effort than that already expended.

5. DESCRIPTION OF CURRENT CAPABILITY

The goal of providing an interactive, nonhierarchical mode of use for ASAN was accomplished in the preliminary prototype version. The present version of the software executing on a Zenith 248 computer is capable of displaying and changing user interface screens at a rapid rate. The software does not enforce a rigid sequence of operations, but rather permits the user to exercise various portions of its capability (e.g., graphics, database access, definition of proposed action, prediction of noise exposure and effects, report production, etc.) at will. All user-specified actions pertain to a default assessment which can also be changed at will.

Source code developed for the preliminary prototype version of ASAN may be found in the Appendices of this report. This code supports the capabilities described in the following sections.

5.1 Screen Description Format Files for User Interface

These files describe the appearance and functioning of the console monitor screens with which users interact to command ASAN. These text files are the input to the U parser that in turn produces C language code that displays the screens.

By convention, each screen is divided into several regions. The screen header (roughly the top four rows) displays a formal title for the screen and the current environmental assessment[2] name (the one available for data entry or analysis) or other information if the actions that can be taken on the screen are assessment-dependent. The middle of the screen (approximately fifteen rows) provides users access to various functions that implement the actions associated with the work that can be done from the screen. The lower portion of the screen (generally four rows) is an area in which actions can be elected to leave the present screen. The bottom row is reserved for error messages and other application code messages to the user. A brief summary of the rationale for these decisions about generic screen layouts is as follows:

1. Consistency in screen design facilitates use of ASAN by minimizing learning time and providing familiar-looking displays for which use habits can be established.
2. A formal title for each screen assists users in keeping track of their progress

[2]The term "environmental assessment" is used internally within ASAN to refer to a set of specifications related to a proposed set of operations. Thus, for example, the term "VR244 assessment" might be used to refer to all information related to assessing the effects of a particular set of aircraft operations on this MTR.

through an environmental assessment, since the basic screen-oriented user interface does not impose any sequential constraints on program use.

3. Constant availability of a description of the current assessment at the top of the screen will assist users in keeping track of which variant of an environmental assessment they are currently operating on.

4. Prioritization of work elements on a screen can be consistently top-to-bottom, left-to-right, so that most frequently performed actions can be accessed with the least cursor motion.

5. Concentration of access to other screens in the lower portion of each screen is consistent with the prioritization of work elements within screens, and minimizes user uncertainty about how to terminate current actions in favor of alternative actions.

6. Provision of a status line at the very bottom of the screen is an unobtrusive way to provide supporting information (e.g., about cursor commands), but can also be used to present important error messages.

Table 5-1 summarizes the principal functions of each screen.

Table 5-1: Summary of Capabilities Accessible from ASAN's Major Screens.

Major User Interaction Screens	
Name of Screen	*Summary of Function*
Title	Announce program name, permit viewing of general information about ASAN, password entry, provide initial access to major screens
Environmental Assessment Status	Summarize assessment in progress, permit switching of assessments
Select Another Assessment	Change current assessment
Data Analysis	Calculate noise exposure, noise effects, access geodata analyses
Environmental Assessment Definition	Add information to current assessment
Select Aircraft and Mission for MTR	Populate current MTR with aircraft and mission types
Mission Specification for an MTR	Describe a mission on current MTR

Table 5-1: continued.

Major User Interaction Screens	
Name of Screen	*Summary of Function*
Define/Modify MTR	Describe characteristics of MTR segments
Flight Parameter Entry	Define operational characteristics of MTR use
Flight Operation Data Entry for MTR	Define numbers of operations on MTR
MTR Definition	Provide organizational information about MTR
MTR Data Entry	Access to MTR specifications
Select Another MTR	Access database of MTRs
Database Housekeeping	View or modify ASAN's permanent databases
View Checklist	View checklists of NEPA-related information
Map Display Control	Select display areas and scales
Map Screen Management	View, compose maps
Point of Contact Database	Interrogate Point of Contact database
Point of Contact Attribute Search	Qualify Point of Contact database search
Display Point of Contact Information	View selected entries in Point of Contact database
Human Effects Citation Search	Interrogate database of references to technical literature on effects of noise on people

Table 5-1: concluded.

Major User Interaction Screens	
Name of Screen	*Summary of Function*
Detailed Display of Retrieved Citations	Display bibliographic information for selected citations
Display Abstracts and Critical Reviews	Display commentary on selected citations
Human Effects Keyword Search	Qualify search of database of citations to noise effects on people
Animal Effects Keyword Search	Qualify search of database of citations to technical literature on effects of noise on animals
Structural Effects Citation Search	Interrogate database of references to technical literature on effects of noise on structures
Structural Effects Keyword Search	Qualify search of database of citations to noise effects on structures
Noise and Sonic Boom Modeling Effects Search	Interrogate database of references to technical literature on generation and propagation of aircraft noise
Legislative Citation Database Search	Interrogate database of legislative and regulatory information about noise
Make a Report	Produce standard and other outputs describing an environmental assessment
View Boilerplate Text	View standard text describing various topics germane to environmental assessments of noise impacts

5.2 SQL Code for Manipulating ORACLE Databases

This code contains the C-language calls to ORACLE with the embedded SQL statements that serve to store and retrieve information in ASAN's various databases.

The connection between ASAN and ORACLE's database servers is transparent to the end user. While ORACLE is a separate process, this is of no concern to the ASAN user. If ORACLE has not been started when ASAN is started, ASAN will start ORACLE first. When ASAN stops, it will leave the machine in its original state. Those processes of ORACLE that ASAN had to start will be taken down, those that were already running will remain.

5.2.1 Code Organization

ASAN code is modularized to produce the most flexible software environment for prototype development. Subprograms are composed of sets of function calls. Each of these functions performs some standard type of task and is used by all ASAN programs that need to have this task performed. This feature provides localization: making a change in the way a task is performed generally requires modification of only one function.

5.2.2 Interfunction Communication and Error Conditions

In ORACLE, as in all ANSI SQL implementations, data manipulation language is processed by a "back end." Success or failure of these interactions is communicated between the back end and the application program through a status code.

To make the SQL interface as transparent as possible, all C-language functions that interface ASAN programs with the back end also return a status code. This feature enables one to write fairly generic functions that can be used by all ASAN modules. In the C language, success (or true) is generally indicated by a non-zero value, while failure (or false) is assigned the value zero. This scheme is inadequate, however, for communication between SQL and application programs.

ASAN adheres, to the extent possible, to the SQL protocol in communicating the status of its database interactions:

- Successful requests return a zero.
- Successful requests with an exception condition return a positive value (e.g., a request for 20 items when there were only 15 to be found).

- Unsuccessful requests return a negative value, which indicates the nature of the error.

All functions that interface with the database are of type integer and return the last meaningful SQL return code. That is, they return the status at the time the error occurred, not the status associated with the success or failure of any subsequent remedial steps.

ASAN displays error conditions on the status line and keeps keeps track of error conditions that might indicate:

1. A problem with the system or database.
2. A user attempt at something that, if allowed to proceed, might lead to a database security violation.
3. A user attempt at something that might, if it happens frequently, indicate a need for further guidance or training in the use of ASAN.

When an error occurs, ASAN stores the time of occurrence and the description of all such errors (often considerably more elaborate than the information shown on the status line). To enable resolution of such problems, this audit file also contains a record of how the system was started, what assessment was being worked on, who the planner was and how the system was stopped.

ASAN will attempt to recover from all errors except security violations, failures during the start of ORACLE processes and a few irreconcilable errors. If a system restart is necessary ASAN will attempt it automatically. ASAN will, however, inform the user that an operation is aborted.

5.2.3 Database Integrity Checking

Modification, insertion and deletion of records in the database make use of the machinery provided by SQL. Integrity of the database is enforced primarily through programmatic checks. In some cases the database schema also performs a check. Such data dictionary checks are usually included for external tables brought into ASAN (e.g., noise and performance data). In this way data integrity is also assured for data brought into ASAN through the ORACLE Data Loader utility program.

In principle, more enforcement could be relegated to the database software. This enforcement was considered of little value in the prototype:

- The diagnostics provided by the data dictionary are usually more directly useful for interactive queries through SQL*Plus than for a programmatic query.

- When data are inserted or updated, data dictionary exceptions will usually cause an automatic ROLLBACK operation to be performed. A programmatic check can avoid having a transaction aborted, making for a superior user interface.
- In the context of ASAN's data dictionary, enforcement of uniqueness and nulls adds mostly unnecessary overhead. The ability of users to access the data outside of ASAN was specifically precluded to assure database integrity.

5.2.4 Screens that Interact Directly with the Database

Because the ASAN user is not forced to perform particular steps in order, ASAN's design contains extensive checking when projecting information into or retrieving data from database tables. As one example of this difficulty, consider how ASAN must respond if a user asks for information to be displayed but the number of rows returned is indefinite. One must plan for the case where more information is contained in the database than can be displayed on a single screen. The user should have the option to pick the information needed or to ask for the next set of rows in the query. But since the user has control over program flow--the user interface is the controlling program--the user cannot, at that point, be permitted to leave the screen and pursue an unrelated activity without first closing the SQL cursors.

Whenever such a condition might occur (e.g., all "multiple choice" screens and those where many screens of input are required before a COMMIT or ROLLBACK can be issued) ASAN will not display the usual set of alternatives at the bottom at the screen, but only give the option to:

1. leave that part of the program without taking any action (or rolling back the transactions entered), or
2. take a limited set of specific actions (including committing the data entry sequence on the database).

Many of these screens also show an indefinite number of options. These screens must first be assembled before they can be displayed (other screens are precompiled). Thus, pressing a button to bring up such a next screen does not transfer control to that screen immediately. There is an intervening call to a program that first assembles the information to be displayed. Since this setup requires noticeably more time, such programs indicate on the status line that some retrieval operation is in progress.

All multiple choice screens (Select Assessment, Select MTR, etc.) use this type of setup machinery. Screens with dynamic (i.e., context sensitive) textblocks are also assembled through a database query. For example, screens that show a context sensitive subset of animals in the taxonomy table or the route description of a specific MTR are assembled when needed.

5.2.5 Programming Style

The program code is not necessarily consistent in the way it handles its interaction with ORACLE. This inconsistency reflects the nature of prototype software. Various approaches were used to test alternatives in terms of storage, execution speed or programming efficiency. Code to be written in the next development phase will reflect the insights gained during the current phase.

All program files contain in-line documentation. Each file starts with a table of contents for that file and a brief description of each function. Each function is preceded by an introductory section that describes what the function does and, when appropriate, what other functions are intimately linked with the function.

Almost all files contain embedded SQL statements. ORACLE's preprocessor is used to convert these statements to standard C-language commands. These statements follow standard SQL syntax and are preceded in the program by the keywords EXEC SQL. Since the preprocessor will truncate lines to 80 characters, some limitations are imposed on the amount of indentation practical in program files. Attempt has been made to make the block structure of the program visible without making the resulting program listings excessively long.

5.3 GRASS-derived Code for Manipulating Graphic Databases

This code contains the ported and modified GRASS code that manages the editing and display of maps in ASAN.

The functions made available to the user by the ASAN GRASS geodata management software implemented for the ASAN demonstration prototype are listed below. These functions are all invoked using a screen-oriented interactive user interface.

- Erase the graphics display.
- Select an area of interest (location and scale) from a set of predefined areas.
- Add a map layer to the graphics display. The name of the desired map layer is typed at the keyboard after viewing a list of all available map layers. Display color selection is performed automatically.
- Remove a map layer from the graphics display. The name of the desired map layer is typed at the keyboard after viewing a list of the map layers currently displayed.
- Display a legend on the graphics display.
- Remove the legend from the graphics display.

- Select a point by specifying its coordinates, either by typing at the keyboard or by passing them from another program.
- Select a region by specifying its border as a series of points.

5.4 Noise Exposure Calculation Code

This code implements the L_{dnmr} calculation procedure of Plotkin, Sutherland and Molino (1987) as expressed in AAMRL's ZROUTE program.. The following steps were taken to provide sufficient information to exercise ASAN's exposure prediction module and print a report:

1. The basic computational part of AAMRL's ZROUTE program, written in BASIC, was translated into C. The build-in data tables were removed and stored in an ORACLE database. The program was then generalized to calculate exposure for any aircraft for which data exist in the database, not just those originally hard-coded into ZROUTE.
2. Additional geometric routines were written so that computations could be made for any arbitrary point on the ground in relation to any arbitrary MTR segment. In keeping with the limited model of ZROUTE, exposure is only nonzero if the perpendicular from the point to the MTR intersects the route segment. No new model was developed to account for exposure from MTRs with curvature.
3. In reasonable approximation to actual practice (and to allow one-for-one comparison with ZROUTE output) calculations assume level flight with fixed aircraft speed and power setting. The mission database in ASAN has been designed, however, so that changes in altitude, power and speed can be accommodated at a later time.
4. Upon entry of a mission of a particular aircraft on an MTR, ASAN will calculate a distance/exposure table comparable to that produced by ZROUTE and will store this information in MTR_EXP_TAB. Since ASAN data entry allows changes in power setting, speed and altitude but ZROUTE does not, ASAN's implementation at this time will calculate exposure based on the aircraft flight parameters specified for the MTR entry point only.
5. The values calculated by ASAN have been verified against parallel runs of ZROUTE and do, in fact, produce identical results.
6. ASAN's report generation module takes the data contained in this table and calculates the noise effects associated with the exposure.

To make effective use of disk storage and to store the most useful data, ASAN does not calculate L_{dn} or L_{dnmr} for a given level of operations. Rather, the program calculates the L_{eq} for a single aircraft on that mission. This approach has several advantages:

- Since one is concerned with seasonality in both operations and exposed population--not usually implemented in previous computer models--one can derive the proper noise metric (such as L_{dn} or L_{dnmr}) for any time period from the operations volume and the L_{eq}. This approach requires storage of only 24 more values (operations during day and night) instead of storing 12 complete sets of calculations or 12 sets of map layers.
- Changes in volume of operations can be evaluated without repeating the most compute-intensive calculations, allowing comparatively fast response times on small computers.
- Noise exposure at large numbers of data points, such as are required for map making, do not have to be stored in both L_{dn} and L_{eq}.

It is anticipated that full-scale calculations (generating contours, detailed analysis reports, etc.) would probably be run as an overnight job or perhaps as a background task in a future multi-tasking operating system environment. Alternatively, a planner might wish to perform calculations in bits and pieces as information is available. Therefore, ASAN will store contributions from each combination of aircraft, mission and MTR and not just a final result. Such partial calculations will allow comparisons to be made among alternative operational scenarios, such as:

- changes in exposure when different aircraft fly a mission;
- changes in exposure due to different power management alternatives;
- relative contribution from various missions flown on an MTR; and
- relative exposure due to operations on different MTRs for areas in the vicinity of several MTRs.

5.5 Noise Effects Calculation Code

This code implements various dosage-effect relationships and table look-up procedures for calculating the prevalence of annoyance associated with noise exposure, interpretation of habitability criteria, prediction of hearing damage risk, and identification of sensitive land uses. None of these relationships was developed specifically for ASAN, nor was any intended to serve as a permanent prediction module.

5.6 Report Production Code

The preliminary prototype version of ASAN does not attempt to produce text that can be inserted directly into a FONSI or other environmental impact assessment document. It provides instead a standard report produced by a prototype report generation module that summarizes the problem definition, noise exposure predictions, and noise effects calculations conducted for each environmental impact assessment. The standard report contains eight sections, as described below.

- Section 1 - Problem definition specifications

  This section echoes the problem definition information supplied by the user, including a printed map. Two variants of the first section (one for MTRs, one for MOAs) are intended.

- Section 2 - Description of predicted noise exposure

  This section summarizes L_{dnmr} calculations for MTRs and frequency of occurrence of sonic booms for MOAs.

- Section 3 - Land use compatibility discussion

  This section reports the results of exercising the habitability interpretation module. It implements a "sensitive land use identification module".

The contents of the next four sections are in part contingent upon the success and severity of effects codes of the noise effects calculation modules.

- Section 4 - Description of inconsequential noise effects

  This section contains a list of noise effects with non-zero success codes and severity of effect code values of 0.

- Section 5 - Description of noise effects of minor importance

  This section contains a mixture of boilerplate and carrier phrases for noise effects with non-zero success codes and severity of effect code values of 1.

- Section 6 - Description of noise effects of considerable importance

  This section contains a mixture of boilerplate and carrier phrases for noise effects with non-zero success codes and severity of effect code values of 2 or greater.

- Section 7 - Description of noise effects not considered in present analysis

  This section warns users of conditions under which insufficient information is available to produce a prediction or of other conditions that preclude explicit consideration of a noise effect.

- Section 8 - References

  This section contains references to support citations made in boilerplate paragraphs.

The report generator interrogates ASAN's database of flight activities and analyzes the contribution from all user-specified noise sources for a given assessment. The noise effects modules themselves generate the specific language that is eventually incorporated in the various sections of the report.

The report generator further considers seasonality in operations by taking into account variability of exposure as appropriate. Boilerplate text (standard wording that is context -independent and hence contained in multiple documents) is appended to each section of the report in which a consequential noise effect is predicted. No text is produced either for inconsequential effects or effects that are irrelevant to the current assessment. Each section of the report is produced by a separate function, so that variant reports can be printed from a set of generic reporting functions. A sample of the output produced by this code may be found in Appendix E of Volume III.

6. REVIEW OF PROTOTYPE DATABASES

Procedures followed in designing and filling the textual and graphic databases on which ASAN operates are described by Kugler, Nicholson, Heeb and Silvati (1987), a summary of which is provided in this section. Details of the procedures followed may be found in Appendix G of Volume III.

The objectives for the database development were to provide the ASAN users with a preliminary set of information and scientific knowledge required in the environmental impact assessment process and to set forth the proposed methodology for the collection, analysis and integration of this data into the ASAN system. The following databases were devised to satisfy these objectives:

- Literature citation database
- Point-of-Contact database
- Legislative database

6.1 Literature Citation Database

The citation database compiles and evaluates the technical literature on effects of sonic booms and subsonic aircraft noise on people, animals, and structures. The database also includes citations to acoustic modeling and aircraft noise studies associated with prediction of exposure to subsonic and supersonic aircraft noise.

The procedures developed to construct the database included creation of a set of rules by which large amounts of published studies could be screened, evaluated and summarized into a form that (a) could be incorporated into ASAN, and (b) could be presented in a form useful to the environmental planner.

The information provided for each publication in the citation database varies depending on the relevance of the publication to the environmental impact assessment process. Figure 6-1 shows the three levels of information provided.

The basic information, which is available for all citations in the database, contains data such as author, publication title, publisher's name and date of publication. The basic information also contains the "suitability rating." This rating, which varies from 1 to 4, is designed to describe how important a particular publication is to the noise and sonic boom impact assessment process.

BASIC INFORMATION

- AUTHOR
- TITLE
- PUBLICATION NAME
- DATE
- SUITABILITY RATING

COMPLETE INFORMATION

- ABSTRACT
- KEYWORDS
- CONTROVERSIALITY RATING

CRITICAL REVIEW

Figure 6-1: Citation Database Publications - Levels of Information Provided.

A rating of 1 means that the information is directly applicable to the environmental impact assessment process and a rating of 4 means that the publication is not relevant.

The complete information contains additional data about each citation. This information is provided only for the more important citations, possibly the ones rated 1 or 2. Many additional items of data are included, including an abstract, keywords, and a "controversiality rating."

The objective of the controversiality rating is to provide the environmental planner with an assessment of how controversial a publication is in light of qualified technical opinion. This is especially important for environmental planners when defending USAF technical assessments in a public forum. This rating will indicate to the planner if the publication is controversial.

Finally, the critical review contains a summary of the strengths and weaknesses of each publication. This summary is prepared only for the most important citations or for very controversial publications and reflects the analysis of an expert in each field. The use of the critical review will provide the environmental planner with a ready interpretation of the value and contents of the specific publication in question.

The scheme for the development and procedure associated with the citation database is discussed in detail in Appendix G of Volume III. To demonstrate this feature in the preliminary prototype version of ASAN, publications were searched, collected and analyzed as shown in Table 6-1.

Table 6-1: Numbers of Entries in Citation Databases.

Citation Database Entries	
Effect Category	*# of Entries*
Human effects entries	630
Animal effects entries	621
Structural effects entries	448

The total number of citations contained in the preliminary prototype version of ASAN is 1,699. About a third of these citations contain complete information. Only a few citations have gone through the critical review process.

The human effects citation database requires the greatest effort to complete in that the technical literature on effects of noise on people consists of roughly 10,000 publications. While it is not anticipated that 10,000 citations will ultimately be included in the human effects database, the review process associated with the first step in the citation database procedure will nonetheless require considerable effort. This review was completed for approximately 2,000 publications during the preparation of the preliminary prototype system. Of these, 630 publications received suitability ratings of 1, 2, or 3.

The entries prepared for the animal effects citation database represent approximately half of the ultimate size of this database. This estimate takes into consideration the completeness of the basic information described in the citation database procedure. The structural effects database contains somewhat more than half of the total number of entries expected.

The future effort in completing the just mentioned databases should be viewed separately for the structural area and the human/animal areas. A separate task under Contract F33615-86-C-0530 dealing with the complete structural effects literature search and review was scheduled for completion in June 1988. The remaining citations in the structural area not currently in ASAN will be added and the review procedures completed by the end of this task.

The efforts required to complete the animal and human effects databases are more difficult to estimate. The major uncertainty is the number of the publications that will be subject to the

full three step review. The third step is especially expensive since it requires that an expert (or multiple experts in the case of a controversial publication) review the contents of a publication in sufficient detail to develop an opinion and write a critical review on the subject. Two to five days' effort for each selected paper is required to accomplish this step. The practice to date has been to limit these reviews to the most important or the most controversial papers, but the ultimate limit may be determined by the availability of funding.

6.2 Point-of-Contact Database

The point-of-contact database contains names, addresses and telephone numbers of individuals or organizations useful to environmental planners in collecting site specific data. The database contains both national and local contacts.

Construction of this database required two separate efforts. The first effort was development of the procedures and screens needed to facilitate data entry. These procedures are detailed in Appendix G of Volume III. The second effort was the collection of actual points of contact. These contacts were limited to those associated with the Sells MOA demonstration site.

6.3 Legislative Database

The legislative database is intended to contain a collection of laws and regulations governing the preparation, content and criteria for environmental impact assessment. As such, the database will contain national, state, and local regulations.

No applicable Arizona regulations governing allowable noise exposure of humans, animals or structures were found for the Sells demonstration site.

7. SUGGESTIONS FOR DEVELOPMENT OF FINAL PROTOTYPE SYSTEM

Although the preliminary prototype version of ASAN demonstrates the concepts of operation described previously, it is neither functionally complete nor seamlessly interfaced. Further procedural work is needed to develop a fieldable version of ASAN. This further effort should be undertaken in three stages.

In the first stage, integration of modules should be completed, the more fragile parts of the preliminary prototype version should be fortified, and a limited expansion of the analytical and computational capabilities of ASAN will be completed. The objective of this stage is to limit reworking of ASAN features to those needed for a meaningful evaluation of the system by a selected group of ("alpha test") environmental planners.

In the second stage, feedback should be obtained from these alpha-test users. The objective of this step is identification of the desirable and undesirable features of ASAN that need modification or enhancement. A summary of the comments of the alpha-test users will be prepared

In the third stage, modifications based on feedback of the alpha-test users will be incorporated. Modifications and extensions of ASAN capabilities may also be made on the basis of results of ongoing exposure and effects studies sponsored by NSBIT.

7.1 Stage 1 - Development of the Final Prototype Version of ASAN

The immediate effort recommended for the prototype can be divided into three major areas as discussed below. Stage 1 will require 9 to 12 months for completion.

7.1.1 Integration

ASAN consists of groups of functions aggregated into major sets of modules: the analytic modules (ASAN), the graphic modules (GRASAN), the report modules (RPTASAN) and the citation modules (CITASAN). Each of these modules is sizable, requiring between 256 and 512 kB of RAM each. Given that MS-DOS cannot make use of more than 640 kB, more is required to integrate ASAN's capabilities than merely stringing these 4 major collections of modules together.

Now that basic capabilities in each area have been demonstrated, the next step is to make the boundaries between these modules transparent to the user. This entails restructuring data storage so that the DGROUP data, stack and heap space segments fit within the 640 kB limitation, and segmentation of the code is such that this block together with the CODE segments and resident processes fits inside the 640 kB boundary.

In systems such as ASAN where large blocks of storage are needed for interprocess communication, this is a nontrivial effort even when code has been developed with this structure in mind. Furthermore, ASAN can be expected to grow as additional models move from development to computer implementation (e.g., structural effects). Thus, the integration must be made sufficiently general and transparent that each subsequent addition will not require additional resegmentation of the program.

This step will also identify requirements of the computer hardware, the MS-DOS (or other) operating system, the C language, the user interface, the SQL data manager and the geodata manager into a formal set of conventions that will comprise the ASAN internal software protocol. Modules adhering to this protocol can be brought under the umbrella of ASAN.

This step is crucial to further development of the software. From the user's perspective, it will provide a seamless integration of all ASAN capabilities. That is, once ASAN is started, the user will not require the services of the operating system to exercise all of ASAN's features.

These technical efforts will result in:

- Adjustments to the interaction between the screen manager software and the control logic.
- Development of protocols for the interaction between the database manager and the analytic code, including the incorporation of expanded schemata into the database and achievement of a high degree of data independence between database structure and application code.
- Development of protocols for the interaction between the geodata manager and the analytic code, including the ability of users to directly access and interact with both analytic and graphic modules without intervening processes.
- ASAN module structure and communication protocol.

7.1.2 Fortification

The preliminary prototype version of ASAN is not consistently robust. Although portions of the code that could corrupt the database were thoroughly tested and checked, in other areas checkout stopped when it became apparent that a particular capability could be demonstrated. Unpredictable results could be produced under some occasions by these portions of ASAN code.

These weaker areas in the program, many of which are known and identified, need to be fortified. A production version of ASAN for alpha testing should be error free in its internal working. It should also present a consistent interface to the user, without displaying unexpected or extraneous messages.

This work entails trapping of error conditions and other exceptions, improving diagnostics supplied to the user, verifying that internal status information is correctly updated and sanitizing interprocess communication. In addition, partial or improperly coded information in some of the data files used for the demonstration prototype needs to be corrected.

The visible payoff for ASAN users will be:

- Information on the screen will accurately reflect the information used by the machine for computation, without conflicting or extraneous prompts or messages.
- ASAN will not be able to cause the operating system to lose control of the processor, as may happen in the preliminary prototype version when status information does not correctly reflect the internal status of the program.

This activity will also result in an overall data-storage architecture and data access procedure that includes both ASAN's text and graphic information.

7.1.3 Expansion of Functionality

Some functions of the preliminary prototype version of ASAN are implemented as a subset of the desired capability, some are implemented in some instances but not in others; and some capabilities have not been implemented at all either because they require the work discussed earlier or because their mathematical models are not yet fully developed (e.g., for certain of the animal and structural effects).

Before the prototype can go into alpha-test evaluation by a selected set of environmental planners it is desirable that some of these capabilities be implemented. These high-priority implementation items are:

- Develop a model to produce continuous MTR noise exposure values so that meaningful computations can be performed for finite flight segments.

- Implement a more useful report generation capability based on these calculations.
- Implement MTR point of interest calculations.
- Implement initial capability using touch screen as a graphic input device.
- Implement ability to show intermediate results for MTR calculations.
- Implement ability to show individual effects calculations.
- Implement some of the probabilistic models for supersonic MOAs.
- Implement ability to print entries in the literature citation database.
- Implement the first versions of structural effects and animal effects modules which are expected to be available within this stage period of performance.

7.2 Stage 2 - Alpha-Test User Evaluation

At the completion of Stage 1, a group of environmental planners will be invited to a hands-on evaluation of the prototype ASAN. This evaluation may be conducted on a one-on-one basis with an ASAN developer working with a planner and exploring all the capabilities of the program, or it could optionally be conducted in a group setting.

The objectives will be to explore the capabilities of ASAN from the perspective of the user. Although an informal job analysis was conducted at the beginning of the effort described in this report, a review of the steps environmental planners take in conducting typical assessments would be valuable.

The results of the alpha testing will be summarized and evaluated, and then used to develop specifications for the final prototype version of ASAN. Alpha testing can be completed in 2 to 3 months.

7.3 Stage 3 - Development of the Final Prototype Version of ASAN

Based on the results of Stage 2, features will be added or modified to reflect the input from the alpha test phase. Additionally, the repertoire of capabilities should be extended so that ASAN can be used for a wider set of problems and by a larger, and possibly less sophisticated, user community.

Development leading to the beta test version would likely include:

- Incorporation of more sophisticated MTR calculations based on the model expected to be published in by the USAF AAMRL.
- Incorporation of supersonic MOA models developed under other NSBIT projects.
- Incorporation of additional structural effects calculations modules which will be available within this period.
- Ability to bring in more or less "standard" machine readable data from outside ASAN, e.g., digital maps.
- Expanded forward geodata query capability (e.g., calculation of percent of population in an area affected and possibly comparisons between various areas) where analytic modules make use of the geodata.
- Expanded backward geodata query capability (e.g., given a point on a map, which activities contribute to the exposure of this area) where the map analysis make use of the analytic modules.
- Incorporation of more extensive text editing and formatting capability so that output from ASAN can be included into formal USAF environmental documents.
- Graphic input/output capability using a complement of peripheral equipment to be determined during the alpha-test phase.

The hardware environment (microprocessor, peripheral equipment, and locus of conduct of specific tasks) should also receive attention as part of the evaluation process. Evolution of microcomputer hardware and the resulting USAF procurement policies and practices are both factors in deciding what the preferred implementation environment will be.

Depending on the complexity of the changes and modifications required by Stage 2 and the expansions just listed, this stage can be completed in a 6 to 9 month period.

References

Date, C.J. (1986) An Introduction to Database Systems, 4th Ed., Reading, MA: Addison-Wesley Publishing Co.

Federal Systems Division, Wang Laboratories Inc. (1986) "Resource Guide for the AMMUS Contract (Air Force Minicomputer Multi-User System)," Contract Number F19630-86-D-0001.

Fidell, S., and Harris, M. (1987) "Design Recommendations for Environmental Planning Aid Software," BBN Report 6342.

Fidell, S., Harris, M., and Reddingius, N. (1988) "Detailed Design Specifications for a Prototype Assessment System for Aircraft Noise (ASAN)," HSD-TR-88-002 (see also BBN Report 6499).

Fidell, S., Horonjeff, R., and Green, D.M., (1981) "Statistical Analyses of Urban Noise," Noise Control Engineer, Vol. 16, No. 2, 75-80.

Galloway, W., (1981) "Assessment of Community Response to High Energy Impulsive Sounds," Report of Working Group 84, Committee on Hearing, Bioacoustics, and Biomechanics, National Research Council, National Academy of Sciences, Washington, D.C.

Harris, M., and Fidell, S. (1987) "Design Specifications for a Preliminary Environmental Planning Aid System," BBN Report 6349.

Kernighan, B.W., and Ritchie, D.M. (1978) The C Programming Language, Englewood Cliffs, NJ: Prentice-Hall, Inc.

Kugler, B.A., Nicholson, B., Heeb, M. and Silvati, L. (1987) "ASAN Database Development", NSBIT Technical Operating Report No. 5.

Martin, J. (1976) Principles of Data-base Management, Englewood Cliffs, NJ: Prentice-Hall, Inc.

ORACLE Database Administrator's Guide (1987) Part Number 3601-V5.1, Belmont, CA: ORACLE Corporation.

Plotkin, K.J., Sutherland, L. and Molino, J. (1987) "Environmental Noise Assessment for Military Aircraft Training Routes," Vol. 2: Recommended Noise Metric, AAMRL-TR-87-001.

Schultz, T.J. (1978) "Synthesis of Social Surveys on Noise Annoyance," J. Acoust. Soc. Amer., 4 (2), 377-403.